Published in Image Processing On Line on 2016–05–05.
Submitted on 2016–02–10, accepted on 2016–04–27.
ISSN 2105–1232 © 2016 IPOL & the authors CC–BY–NC–SA
This article is available online with supplementary materials,
software, datasets and online demo at
https://doi.org/10.5201/ipol.2016.165

Directional Filters for Cartoon + Texture Image Decomposition

Antoni Buades, Jose-Luis Lisani

Universitat Illes Balears, Spain ({toni.buades, joseluis.lisani}@uib.es)

Communicated by Pablo Musé *Demo edited by* Jose-Luis Lisani

Abstract

We present in this article a detailed analysis and implementation of the cartoon+texture decomposition algorithm proposed in [A. Buades, J.L. Lisani, "Directional filters for color cartoon + texture image and video decomposition", Journal of Mathematical Imaging and Vision, 2015]. This method follows the approach proposed by [A. Buades, T. Le, J-M. Morel, L. Vese, "Cartoon+Texture Image Decomposition", IPOL 2011], based on low/high-pass filtering, but replaces the isotropic filters by a bank of low-pass directional filters. The cartoon image is obtained by filtering in the direction that leads to the largest local total variation rate reduction. This permits to improve the performance of the decomposition near image discontinuities, where an halo effect was produced by the previous method.

Source Code

The source code and an online demo are accessible at the IPOL web page of this article[1].

Keywords: cartoon; texture; filtering; directional filters

1 Introduction

The cartoon+texture problem refers to the problem of decomposing a digital image into a cartoon (piecewise-smooth) component and a textural (oscillatory) part. More precisely, if we denote the image by f[2], we are interested in decomposing f into two components $f = u + v$. The image u is the cartoon or geometric component of f while v is the textured component, which should contain essentially the noise and the texture.

We propose a non-linear filter that decomposes the image into a geometric and a oscillatory part depending on a single scale parameter. The use of non-linear filters for image decomposition was

[1] https://doi.org/10.5201/ipol.2016.165

[2] A grayscale (resp. color) image is represented by a function $f : (x, y) \to I\!R$ (resp. $I\!R^3$), where Ω is an open subset of $I\!R^2$, typically a rectangle or square. The image is defined on a continuous domain by interpolation of the values on a discrete set of pixels.

introduced in [3]. The algorithm in [3] computes a non-linear low-high frequency decomposition by using isotropic filters. We illustrate that the use of these filters produces halo effects near strong edges of the image.

In [5] we introduced a new algorithm using directional filters improving the decomposition in the proximity of edges. In this article we give a complete account of the algorithm and display several results for different values of its only parameter.

The article is organized as follows: Section 2 introduces isotropic non linear filter pairs for image decomposition. The new method is described in Section 3; several results of the proposed method are displayed in Section 4; finally some conclusions are provided in Section 5.

2 Cartoon+Texture with Isotropic Non-Linear Filters

The cartoon+texture decomposition $f = u + v$ is analogous to the classical signal processing low-pass/high-pass filter decomposition. However, the cartoon part of an image actually contains strong edges, and therefore all frequencies, up to the high ones, while a texture can also contain middle and high frequencies. Thus, linear decomposition algorithms cannot make a clear cut separation between cartoon and textures. As a result, edges are blurred in the cartoon part while some texture is still present. Y. Meyer [9] proposed to solve the problem using a variational formulation containing two norms: the right decomposition is the one where the cartoon part u has minimal total variation while the oscillatory component has a minimal norm in a dual space of BV. Many variants proposed in the literature consider various functional spaces for the textural part. In particular, strong mathematical geometric arguments are put forward in favor of the TV-L^1 model [6, 8]. An extensive mathematical analysis of Meyer's model in a bounded domain is performed in [1] while a model classification can be found in [2].

In [3, 4] a fast approximate solution was proposed to the original variational problem obtained by applying a non-linear low-pass/high-pass filter pair. A local indicator was built to decide at each point x whether it belongs to a textural region or to a cartoon region. The main characteristic of a cartoon region is that its total variation does not decrease by low-pass filtering. The main characteristic of a textured region is its high total variation due to its oscillations. This total variation decreases very fast under low-pass filtering. Formalizing these remarks the *local total variation* (LTV) at x was defined

$$LTV_\sigma(f)(x) := L_\sigma * |Df|(x), \tag{1}$$

where L_σ is a low-pass filter (a Gaussian kernel) with standard deviation σ.

The *relative reduction rate of LTV* was defined by a function $x \mapsto \lambda_\sigma(x)$, given by

$$\lambda_\sigma(x) := \frac{LTV_\sigma(f)(x) - LTV_\sigma(L_\sigma * f)(x)}{LTV_\sigma(f)(x)}, \tag{2}$$

which gives the local oscillatory behavior of the function f. If λ_σ is close to 0 then it means that there is little relative reduction of the local total variation by the low-pass filter. If instead λ_σ is close to 1, the reduction is important, which means that the considered point belongs to a textured region. Thus, a fast nonlinear low-pass and high-pass filter pair was proposed, defined by weighted averages of f and $L_\sigma * f$ depending on the relative reduction of LTV_σ

$$\begin{aligned} u(x) &= w(\lambda_\sigma(x))(L_\sigma * f)(x) + (1 - w(\lambda_\sigma(x)))f(x), \\ v(x) &= f(x) - u(x), \end{aligned} \tag{3}$$

where $w(x) : [0, 1] \to [0, 1]$ is an increasing function that is constant and equal to zero near zero and constant and equal to 1 near 1. The soft threshold function w was defined by

$$w(x) = \begin{cases} 0 & x \leq a_1 \\ (x - a_1)/(a_2 - a_1) & a_1 \leq x \leq a_2 \\ 1 & x \geq a_2 \end{cases}, \tag{4}$$

where the parameters a_1 and a_2 were respectively fixed to 0.25 and 0.5. If $\lambda_\sigma(x)$ is small, the function f is non-oscillatory around x and therefore the function is BV (or cartoon) around x. Thus $u(x) = f(x)$ is the right choice. If instead $\lambda_\sigma(x)$ is large, the function f is locally oscillatory around x and locally replaced by $(L_\sigma * f)(x)$.

Figure 1 displays the results of the filter pair for $\sigma = 1.5$. In this figure, and throughout the paper, the texture image is displayed by linearly mapping the $[-20, 20]$ range to $[0, 255]$ and saturating values outside this range.

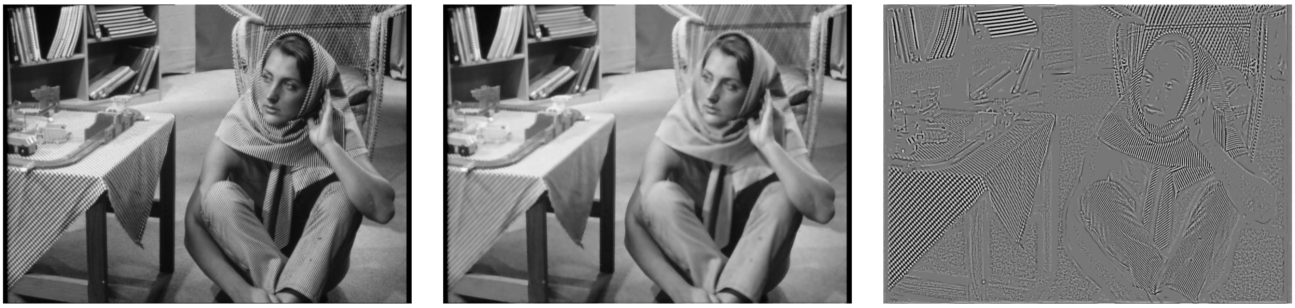

Figure 1: Decomposition of Barbara gray image (left) in cartoon (center) and texture (right). The result was obtained with the non-linear filter pair using L_σ ($\sigma = 1.5$) proposed in [3].

3 Directional Filters for Cartoon+Texture

The filter pair proposed in [3] efficiently separates the smooth parts of the image from the texture components. However, as shown in Figure 2 in the proximity of edges the cartoon and texture are not well separated and the noise and micro-texture are kept in the cartoon part. The reason is that the value of LTV_σ in these regions depends mostly on the sharpness of the nearby edge, which doesn't change too much after the low-pass filtering. Therefore, the value of λ_σ is small and $u \approx f$ in those pixels.

We propose in this section a new approach that overcomes this problem. This approach is based on the use of a bank of low-pass directional filters L_{σ,θ_i}, that replace the L_σ isotropic filter. Let us define $L_{\sigma,0}$ as

$$L_{\sigma,0}(x, y) = \begin{cases} CL_\sigma(x, y) & x \geq 0 \\ Ce^{-\frac{x^2}{2\alpha^2}} L_\sigma(x, y) & x < 0 \end{cases}, \tag{5}$$

where L_σ is a low-pass filter (Gaussian kernel) with standard deviation σ and C is a normalization constant that guarantees that $L_{\sigma,0}$ integrates to 1. $L_{\sigma,0}$ is identical to L_σ for positive values of the x coordinate and decreases exponentially for negative values of x. The spread of the decreasing function is controlled by α, which has been fixed to 0.75 in all of our experiments.

The bank of filters L_{σ,θ_i} is then defined as a set of rotated versions of $L_{\sigma,0}$ around its center. The set of rotation angles is $\{\theta_i = i \cdot \theta_s, i = 0, \cdots, \lfloor \frac{360}{\theta_s} \rfloor\}$, where θ_s is the angular step (fixed to $\theta_s = 8^o$ in all of our tests). Figure 3 displays an example of directional filter, for $\theta_i = 45^o$.

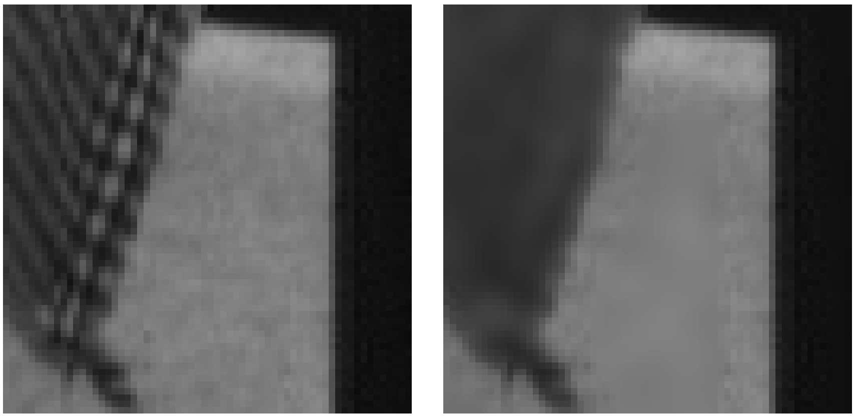

Figure 2: Zoom on the cartoon part of Figure 1 for the non-linear filter decomposition in [3]. Left, original image. Right, cartoon result. Due to the use of an isotropic kernel, the noise and micro-texture are kept in the cartoon part in the proximity of edges (halo effect).

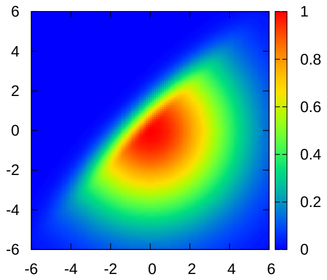

Figure 3: Example of directional filter used for cartoon + texture classification.

The relative reduction of the local total variation rate λ_σ is now defined as

$$\lambda_\sigma(x) = max\{\lambda_{\sigma,\theta_i}(x), i = 0, \cdots, \lfloor \frac{360}{\theta_s} \rfloor\}, \tag{6}$$

where

$$\lambda_{\sigma,\theta_i}(x) := \frac{LTV_{\sigma,\theta_i}(f)(x) - LTV_{\sigma,\theta_i}(L_{\sigma,\theta_i} * f)(x)}{LTV_{\sigma,\theta_i}(f)(x)}, \tag{7}$$

and

$$LTV_{\sigma,\theta_i}(f)(x) := L_{\sigma,\theta_i} * |Df|(x). \tag{8}$$

With this definition, for a pixel x in the proximity of an edge oriented along the θ_i direction, $\lambda_{\sigma,\theta_i}(x)$ shall not be affected by the edge and therefore, if the pixel belongs to a texture, the value of $\lambda_{\sigma,\theta_i}(x)$ shall be high. In consequence, $\lambda_\sigma(x)$ will also be high, indicating that the pixel belongs to the textural component of the image.

Figure 4 displays the values of λ_σ, for increasing values of σ, at various locations of the image Barbara. The left column shows the selected pixel, the second column displays λ_σ computed with the isotropic kernel (Equation (2)) and the third column displays the new value of λ_σ (computed with Equation (6)). We observe that in regions far away from an edge (either smooth or texture regions, rows 1 and 2) the original and new values are very similar. However, in the proximity of edges (rows 3 and 4) the new values are higher. As a consequence, those pixels shall be considered as texture by the proposed algorithm and therefore filtered out from the cartoon component.

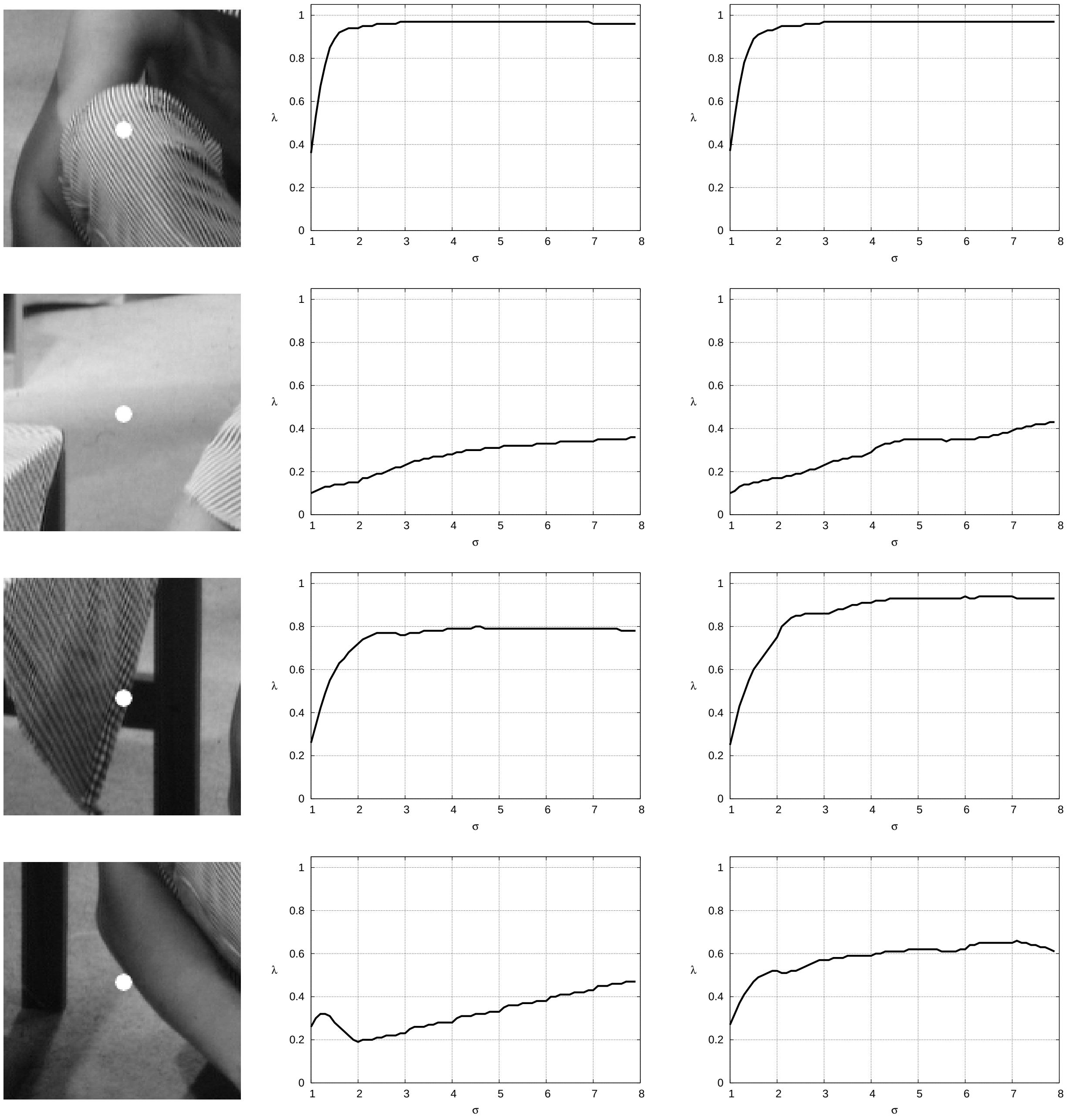

Figure 4: Display of plots $\lambda_\sigma(x)$ for several pixels in the Barbara gray image and different σ. Second column, isotropic filter. Third column, directional filters.

The cartoon and texture images are recovered using Equation (3). For color or multispectral images $f = (f_1, \cdots, f_n)$, the algorithm is applied on each channel, although the oscillatory indicator function is computed as

$$LTV_{\sigma,\theta_i}(f)(x) := L_{\sigma,\theta_i} * |Df_1|(x) + \cdots + L_{\sigma,\theta_i} * |Df_n|(x). \tag{9}$$

That is, all channels collaborate at the same time in the identification of oscillatory parts of the image. The final weighted average uses for each pixel the same weight for all channels, leading to a coherent color image decomposition. For variational methods the fidelity and regularity terms of multivalued functions rewrite as the sum of the same terms in each channel, which is equivalent to the minimization of the variational formulation independently for each channel.

The method keeps the scale σ of the texture as the only method parameter. There is no clear definition of what a cartoon-texture decomposition should be. We argue that such a definition cannot be proposed without fixing a certain scale or delivering a set of cartoon images for a complete range of scales. A textured pattern can be perceived as smooth when viewed from a long distance (e.g. the striped pants in Barbara image, Figure 1-left); conversely, an apparently smooth object can display textural details when viewed at a close range (e.g. the carpet in Barbara image, Figure 2-left). That is, scale is naturally linked to a cartoon-texture definition. For a fixed scale, the proposed cartoon decomposition is invariant to contrast, that is, no parameter is necessary if image contrast is modified. This requirement should be extensive to any other decomposition method, for the current approach it led to comparing the relative decreasing of the total variation and not its absolute one.

The whole method is described in Algorithm 1. Remark that in the description of the algorithm an optional step has been added: the possibility of including the isotropic Gaussian filter in the filter bank. The reason is that in uniform regions of the image (i.e. far from an edge) the results of the original method in [3] were already good. By adding the Gaussian kernel to the bank of filters we ensure that our results will be at least as good as those obtained with the isotropic filter pair. In all the tests displayed in the next section, and also in the online demo, we have used this option.

Cartoon+Texture Decomposition of Video Sequences. In [5] it is argued that the proposed method is better suited for frame by frame processing of video sequences than other state of the art methods [6], [7] and [10]. Two factors explain this performance: first, the proposed method slightly smoothes the edges as an anisotropic filter would do, making them more coherent in time. For the rest of methods the shape of the boundaries is more dependent on small oscillations or noise at the contours. A second reason is that our algorithm behaves as a Gaussian convolution far from object boundaries and in presence of isotropic textures. This convolution is actually translation invariant and thus more suitable for frame by frame processing. This is not the case for variational or thresholding methods.

4 Results

Figure 5 compares the results of the isotropic and directional filters described in the previous sections. Both methods have been applied on the same image and some details are displayed. We observe that artifacts near the edges have been reduced.

In the rest of this section we show the effect, for the images in Figure 6, of increasing the scale parameter σ in the obtained cartoon and texture images. The results are shown in figures 7 to 10. As expected, when σ increases textural details are added to the texture image, since their relative size with respect to the scale parameter decreases and they are no longer assigned to the cartoon image by the low-pass filter. These figures illustrate at which scale patterns become oscillatory.

5 Conclusions

We have presented a new filter based approach for image decomposition that avoids artifacts of previous filter based algorithms. Our method replaces the isotropic filters used in previous approaches by a bank of low-pass directional filters. The cartoon image is obtained by filtering in the direction that leads to the largest local total variation rate reduction. This permits to improve the performance of the decomposition near image discontinuities, where an halo effect was produced by previous methods. Several results are presented and various values of the parameter have been tested.

Algorithm 1: Cartoon+Texture Decomposition using Directional Filters

```
input  : Original image $f$, scale parameter $\sigma$
output: Cartoon and texture images $u$, $v$
begin
    //Create bank of directional filters. $\theta_s = 8^o$ (angular step)
    $\mathcal{L}_\sigma \leftarrow \emptyset$        // Initialization
    $L_{\sigma,0} \leftarrow \begin{cases} CL_\sigma(x,y) & x \geq 0 \\ Ce^{-\frac{x^2}{2\alpha^2}} L_\sigma(x,y) & x < 0 \end{cases}$        ($\alpha = 0.75$, $L_\sigma$: Gaussian kernel)
    for $i = 0, \cdots, \lfloor \frac{360}{\theta_s} \rfloor$ do
        $\theta_i \leftarrow i \cdot \theta_s$        //set of angles
        $L_{\sigma,\theta_i} \leftarrow$ rotate $L_{\sigma,0}$ $\theta_i$ degrees around its center
        $\mathcal{L}_\sigma \leftarrow \mathcal{L}_\sigma \cup L_{\sigma,\theta_i}$        // Add kernel to filter bank
    //Optional: add isotropic kernel to filter bank
    $\mathcal{L}_\sigma \leftarrow \mathcal{L}_\sigma \cup L_\sigma$
    //Compute local directional total variation rates
    for $K \in \mathcal{L}_\sigma$        //For each kernel in the filter bank do
        //Compute Local Total Variation with respect to kernel $K$, at each pixel $x$
        $LTV_K(f)(x) \leftarrow \begin{cases} K * |Df|(x) & \text{gray level images} \\ K * |Df_1|(x) + \cdots + K * |Df_n|(x) & \text{color images} \end{cases}$
        //Local Total Variation Rate for kernel $K$
        $\lambda_K(x) \leftarrow \frac{LTV_K(f)(x) - LTV_K(L_K * f)(x)}{LTV_K(f)(x)}$
    //Total Local Variation rate at each pixel $x$
    $\lambda_\sigma(x) \leftarrow max\{\lambda_K(x), K \in \mathcal{L}_\sigma\}$
    //Optimum kernel at pixel $x$
    $K^* \leftarrow \text{argmax}_{K \in \mathcal{L}_\sigma}\{\lambda_K(x)\}$
    //Cartoon component
    $u \leftarrow w(\lambda_\sigma)(K^* * f) + (1 - w(\lambda_\sigma))f$        //$w$ defined as in Equation (4)
    //Texture component
    $v \leftarrow f - u$
```

Future research might potentially investigate the automatic adjustment of the scale parameter σ. As observed in Figure 4, the value of $\lambda_\sigma(x)$ in textured regions tends to a constant value as σ increases. This hints to the possibility of automatically setting σ to the smallest value attaining this constant. In cartoon regions however the value of $\lambda_\sigma(x)$ follows an increasing trend and further research should be conducted to establish the optimal value of σ. Observe that this line of research would lead to several local scale parameters, instead of the single global one used so far.

Acknowledgment

The authors were supported by the Ministerio de Ciencia e Innovación (Spain) under grants TIN2011-27539 and TIN2014-53772.

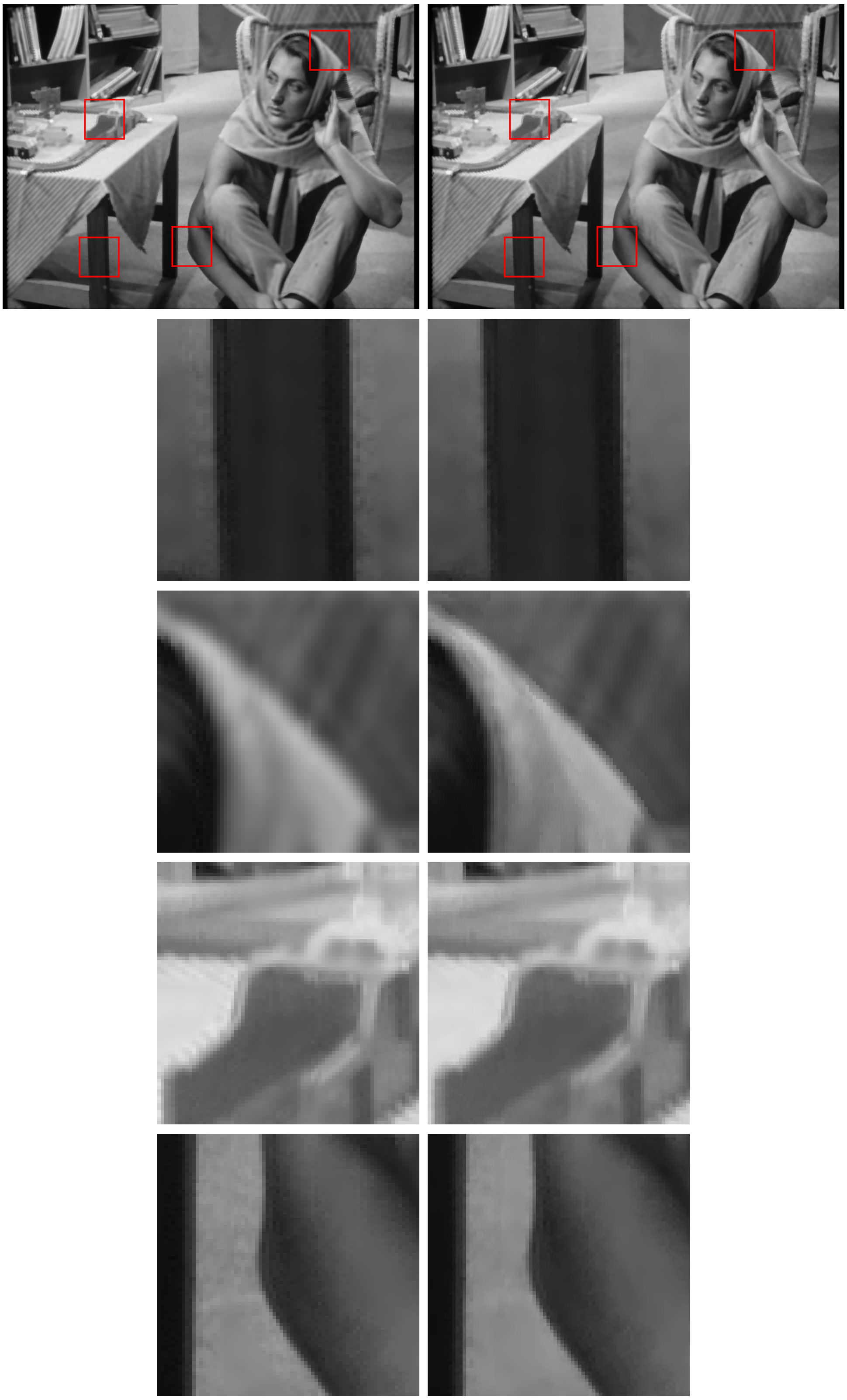

Figure 5: Results on Barbara gray image of the methods described in sections 2 and 3 with the same scale parameter ($\sigma = 3$). Left, isotropic filter. Right, directional filter. Bottom, details corresponding to the red squares marked on the top images. Remark that the halo effect in the proximity of edges has been reduced with the proposed method.

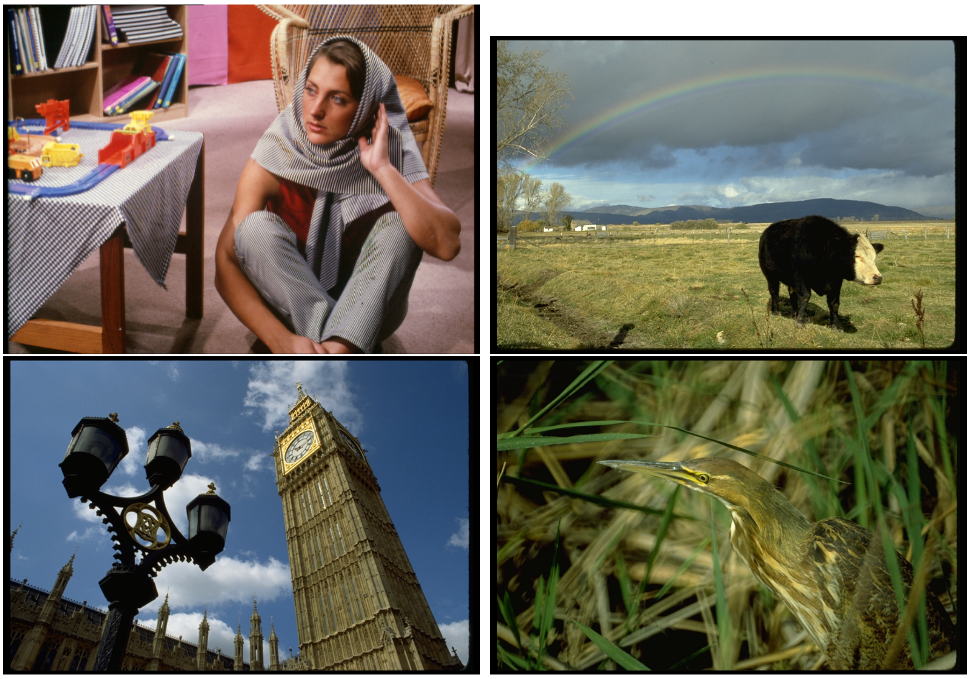

Figure 6: Original images used to test the proposed algorithm.

Image Credits

Standard test image.

Kodak database.

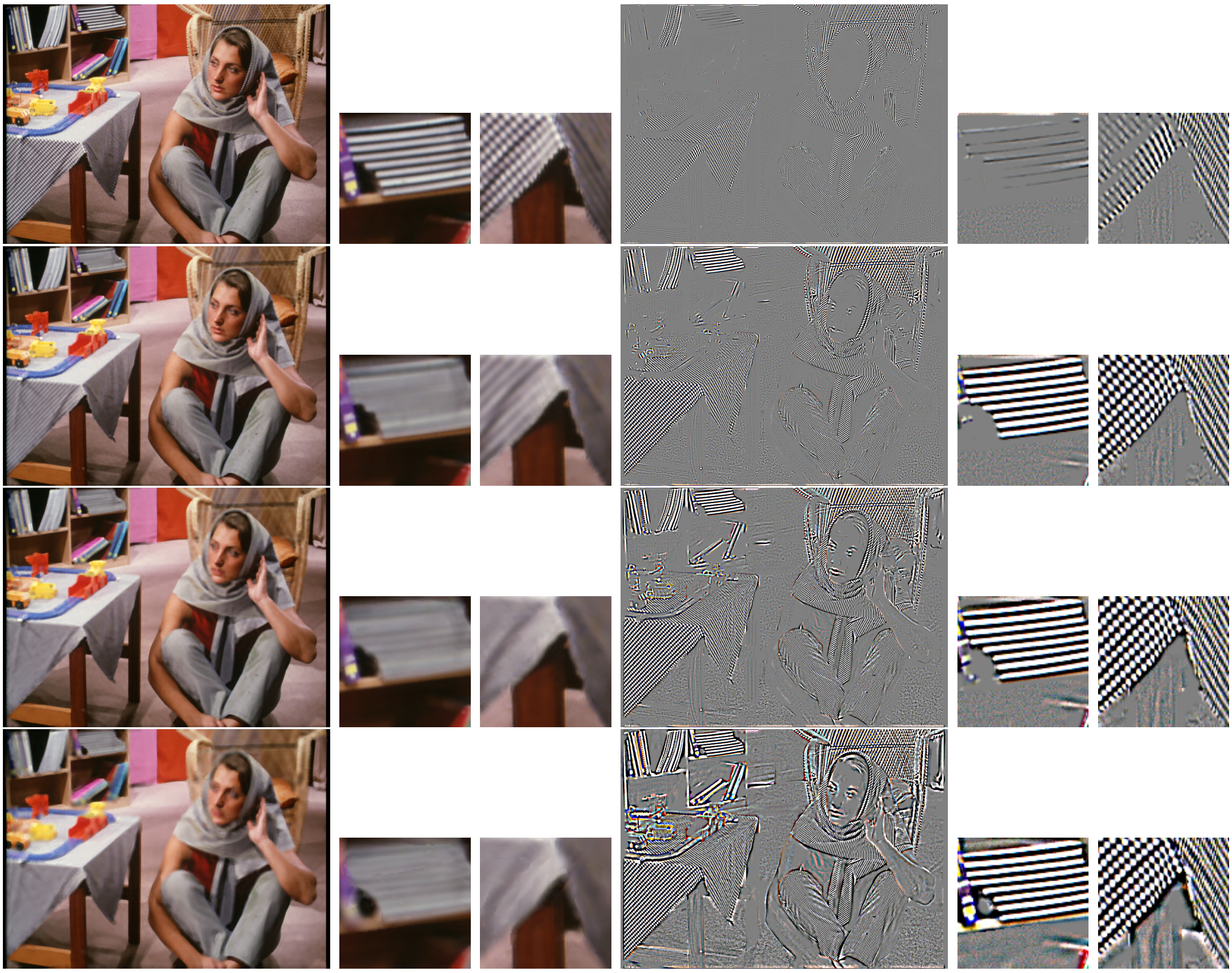

Figure 7: Cartoon decomposition obtained with the proposed algorithm for different values of σ. From top to bottom $\sigma = 2, 3, 4, 6$. When σ increases textural details are added to the texture image, since their relative size with respect to the scale parameter decreases and they are no longer assigned to the cartoon image by the low-pass filter.

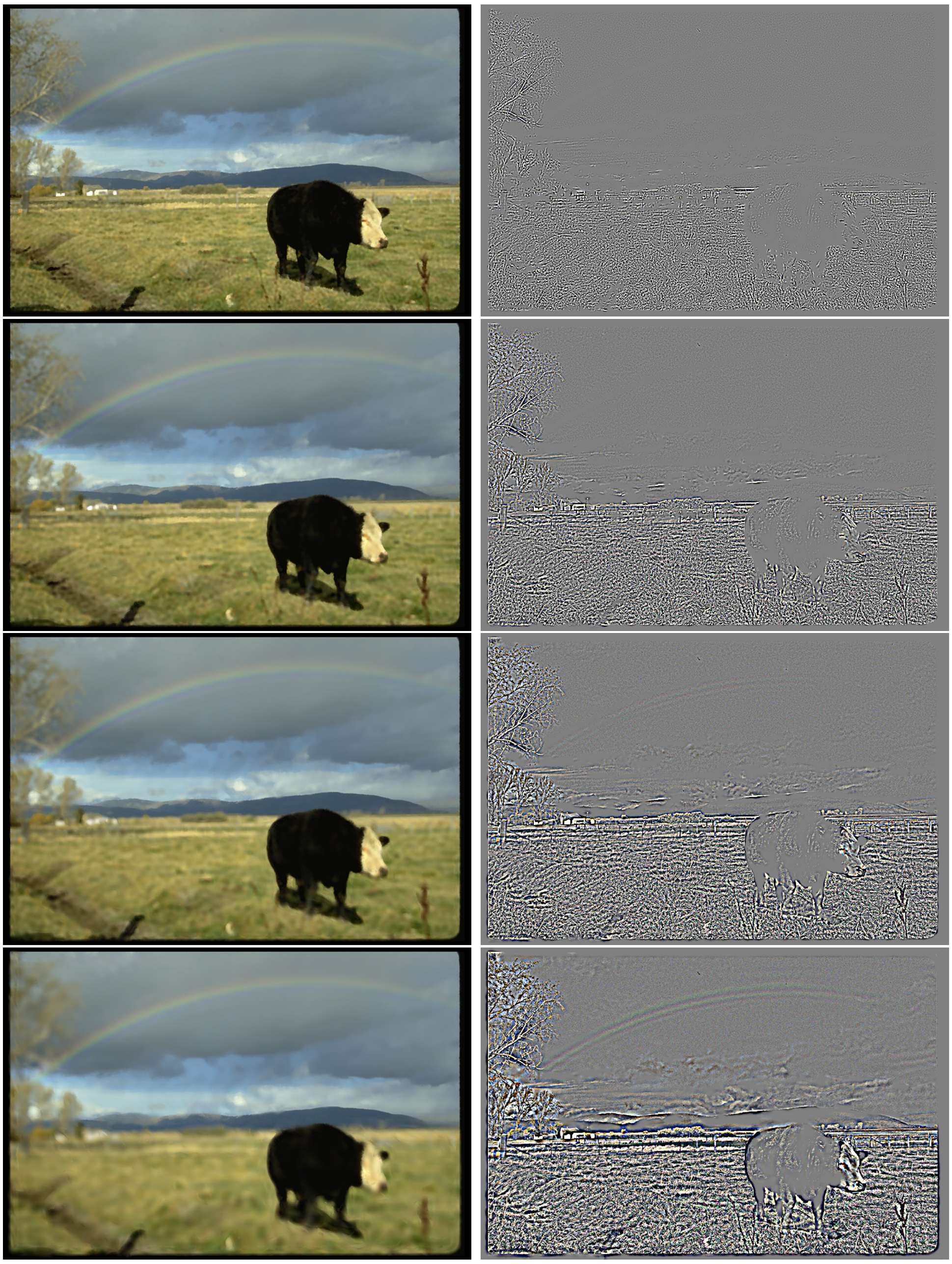

Figure 8: Cartoon decomposition obtained with the proposed algorithm for different values of σ. From top to bottom $\sigma = 2, 3, 4, 6$.

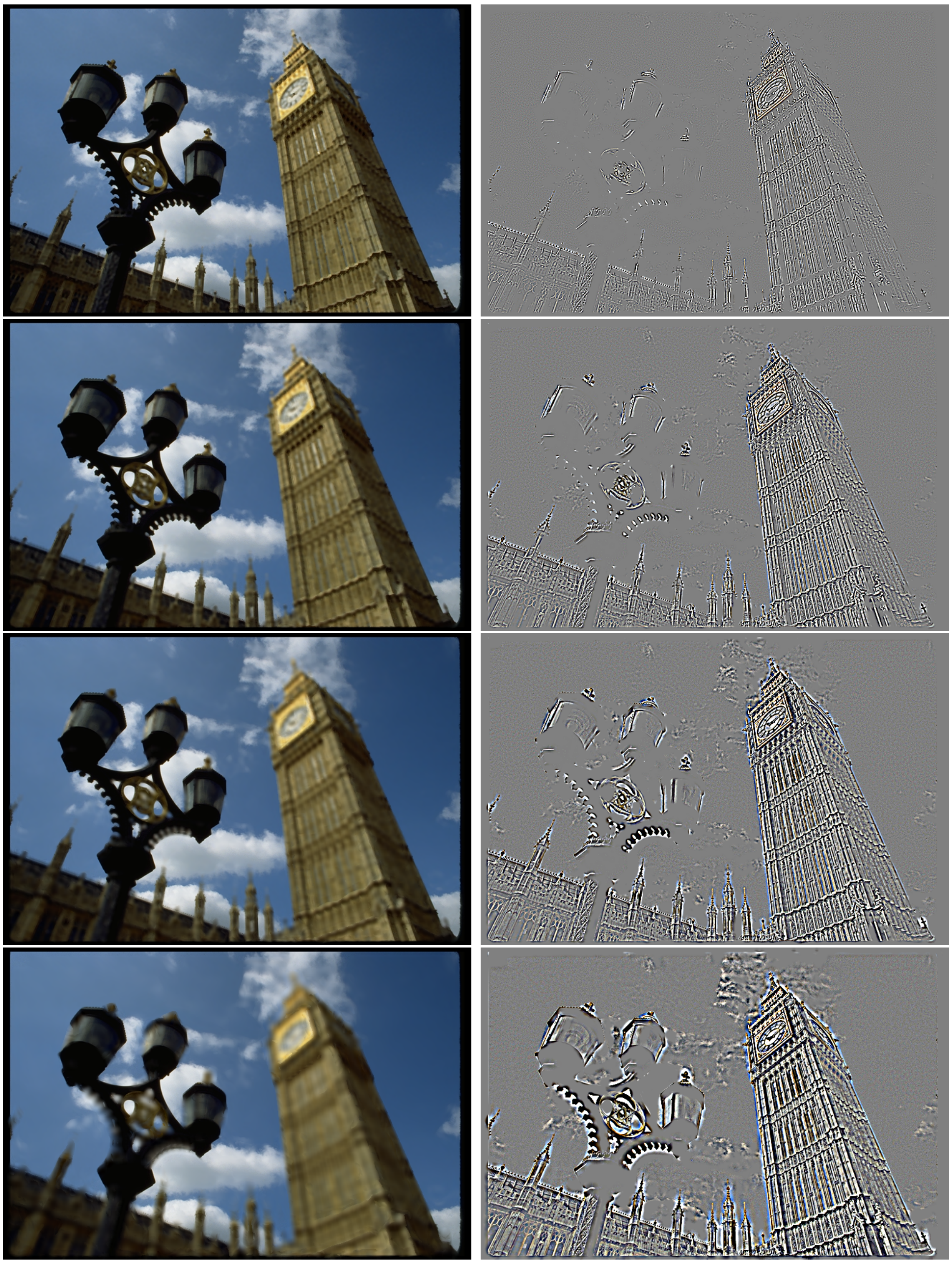

Figure 9: Cartoon decomposition obtained with the proposed algorithm for different values of σ. From top to bottom $\sigma = 2, 3, 4, 6$.

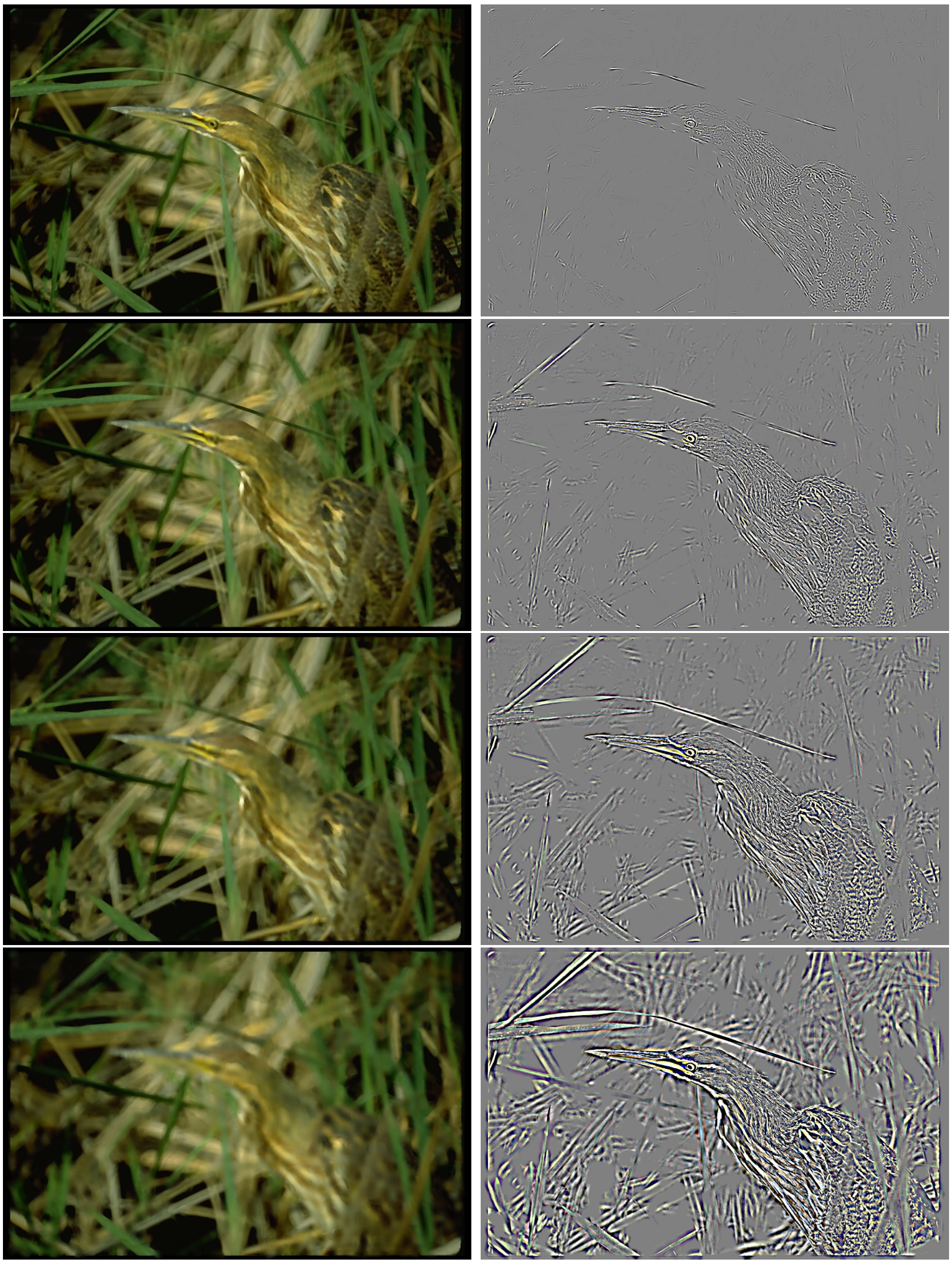

Figure 10: Cartoon decomposition obtained with the proposed algorithm for different values of σ. From top to bottom $\sigma = 2, 3, 4, 6$.

References

[1] G. Aubert and J.F. Aujol, *Modeling very oscillating signals. Application to image processing*, Applied Mathematics and Optimization, 51 (2005), pp. 163–182. http://dx.doi.org/10.1007/s00245-004-0812-z.

[2] J.F. Aujol, G. Gilboa, T. Chan, and S. Osher, *Structure-texture image decomposition - modeling, algorithms and parameter selection*, International Journal of Computer Vision, 67 (2006), pp. 111–136. http://dx.doi.org/10.1007/s11263-006-4331-z.

[3] A. Buades, T.M. Le, J.-M. Morel, and L.A. Vese, *Fast cartoon + texture image filters*, IEEE Transactions on Image Processing, 19 (2010), pp. 1978–1986. http://dx.doi.org/10.1109/TIP.2010.2046605.

[4] A. Buades, T. Le, J-M. Morel, and L. Vese, *Cartoon+Texture Image Decomposition*, Image Processing On Line, 1 (2011). http://dx.doi.org/10.5201/ipol.2011.blmv_ct.

[5] A. Buades and J.L. Lisani, *Directional filters for color cartoon + texture image and video decomposition*, Journal of Mathematical Imaging and Vision, 55 (2016), pp. 125–135. http://dx.doi.org/10.1007/s10851-015-0617-5.

[6] V. Duval, J.F. Aujol, and Y. Gousseau, *The TVL1 model: a geometric point of view*, Multiscale Modeling and Simulation, 8 (2009), pp. 154–189. http://dx.doi.org/10.1137/090757083.

[7] M-J Fadili, J-L Starck, J. Bobin, and Y. Moudden, *Image decomposition and separation using sparse representations: an overview*, Proceedings of the IEEE, 98 (2010), pp. 983–994. http://dx.doi.org/10.1109/JPROC.2009.2024776.

[8] V. Le Guen, *Cartoon + Texture Image Decomposition by the TV-L1 Model*, Image Processing On Line, 4 (2014), pp. 204–219. http://dx.doi.org/10.5201/ipol.2014.103.

[9] Y. Meyer, *Oscillating patterns in image processing and nonlinear evolution equations*, University Lecture Series, 22. American Mathematical Society, 2001. ISBN:0821829203.

[10] S. Ono, T. Miyata, and I. Yamada, *Cartoon-texture image decomposition using blockwise low-rank texture characterization*, IEEE Transactions on Image Processing, 23 (2014), pp. 1128–1142. http://dx.doi.org/10.1109/TIP.2014.2299067.